Using Improved YOLOv5 Model to Detect Volume for Logs in Log Farms

Xianqi Deng[1], *Jianping Liu*[1, 2*], *Cheng Peng*[3], *Yingfei Wang*[1]

[1] *School of Computer Science and Engineering, North Minzu University, China*
[2] *The key Laboratory of Images and Graphics Intelligent Procession of State Ethnic Affairs Commission,*
North Minzu University, China
[3] *College of Medicine, University of Florida, United States*
dengxianqi2001@outlook.com, liujianping01@nmu.edu.cn, c.peng@ufl.edu, 20217426@stu.nmu.edu.cn

Abstract

In this paper, we propose a new computer vision model called SE-YOLOv5-SPD for counting the number of log ends in large wood piles in log farms. This task traditionally requires a lot of manpower and previous computer vision methods struggle to detect logs in low pixels and small objects in images. Our model is based on the YOLOv5 model and incorporates the Squeeze-and-Excitation Networks (SENet) attention module and SPD-Conv (Space-to-Depth Convolution) module to improve accuracy. We also compare the performance of the SE attention module and SPD-Conv module to the CBAM attention module and Focus module using the SE-YOLOv5-SPD model. Results show that the SE-YOLOv5-SPD model can achieve excellent results of 0.652 in $mAP_{50:95}$, 0.912 in mAP_{50}, 0.968 in Precision, and 0.864 in Recall in a low-resolution environment with interference, which is significantly better than other models. Our findings indicate that the SE-YOLOv5-SPD model is a promising solution for counting the number of log ends in wood piles.

Keywords: YOLOv5, Logs detection, Squeeze-and-Excitation Networks, SPD-Conv

1 Introduction

Under the background of the targets of carbon peak and carbon neutrality, the intelligent utilization of forest resources is one of the effective ways to save energy and reduce emissions. The forest occupies a vital position, so it is always a great challenge to make rational use of forestry resources such as logs. In this paper, we propose to use a deep learning approach to solve the problem of counting the number of logs in some log farm scenarios concerning the resource management of logs. In daily wood processing operations, counting stacked wood, quality identification, and managing incoming and outgoing storage are crucial tasks before the wood is processed. However, traditional manual inspection has challenges such as high cost, low efficiency, and high error rate. Using the computer vision method to solve the problem of detecting logs is more suitable. Nevertheless, two problems need to be addressed: detecting logs in low-pixel images and detecting small-sized objects at the image's edges. These problems are prevalent in log farm scenarios.

With the development of computer vision, object detection technology can be divided into two categories: machine learning-based and deep learning-based methods. In terms of machine learning methods. Chiryshev used random forest-based statistical methods for log counting [1]. Gutzeit et al. [2] used PWL-Haarcascades (Post Processed Wood Log Haarcascades) model to detect the logs and used KD-NN-A (KD-Tree based Neural Networks Adaptive) model and LSGMC model to segment the logs. In terms of deep learning, researchers have introduced many excellent deep learning models for similar log detection problems, such as the SSD (Single Shot MultiBox Detector) model [3-4], and the YOLOv4-Tiny model [5]. The YOLO family of models was first proposed by Redmon et al. [6] as the YOLO (You Only Look Once) model. And then developed by Redmon et al. [7] and Bochkovskiy et al. [8] to the YOLOv3 model, and YOLOv4 model. Finally, Glenn et al. [9] proposed the YOLOv5 model which is used in this paper.

The YOLOv5 model is one of the most popular object detection models today, with excellent detection, high modularity, and high scalability. YOLOv5 and its improved algorithm are widely used in similar log detection problems, such as in detecting apples on tree branches, field flat jujube, tomato leaves virus, invasive plant seeds, and wheat ears [10-14].

However, the previous computer vision method has poor performance in low-pixel images and can not detect the small-size objects in the image. Although, some existing approaches are able to enhance the model's ability to detect targets, such as attention mechanisms [15-16]. Alternatively, the SPD-Conv (Space to Depth Convolution) module proposed by Sunkara et al. [17] can improve detection for low pixels and small targets. In this paper, we propose to combine the SE attention mechanism and SPD-Conv module with the YOLOv5 model. The experimental results of this paper show that using the attention mechanism alone or the SPD-Conv module alone does not improve the detection logs of the model, but the detection results of combining the two are excellent.

As described above, YOLOv5 has been widely used in similar log detection problems. At the same time, the improved YOLOv5 model based on the SE (Squeeze-and-Excitation Networks) attention mechanism and SPD-Conv module also achieves excellent generalization ability [5, 12]

*Corresponding Author: Jianping Liu; E-mail: liujianping01@nmu.edu.cn

DOI: 10.53106/160792642023122407002

and handles low pixel and small-size objects problems [17].

In this paper, we aim to address the challenges of log detection in low-pixel images and the detection of small-size objects by introducing the SE attention mechanism and SPD-Conv module into the YOLOv5 model. To achieve this, we propose the SE-YOLOv5-SPD log detection model, which is illustrated in Figure 1.

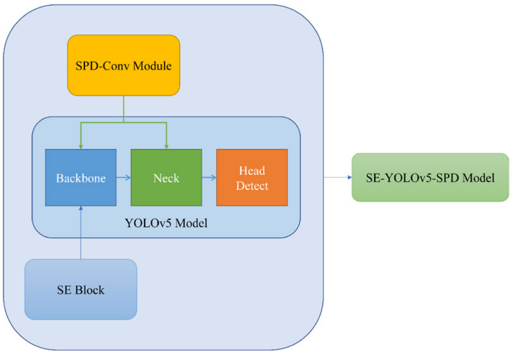

Figure 1. SE-YOLOv5-SPD structure

Note. The SENet Block is added to YOLOv5 Backbone, and SPD-Conv is added to YOLOv5 Neck.

Firstly, we use the YOLOv5 model (Glenn et al., 2022) as the base structure of our proposed model.

Secondly, as shown in Figure 1, we improved the YOLOv5 model by introducing the following two modules: (1). We replace the Conv module [9] with the non-strided convolution [9] and add the SPD-Conv module [17] before the C3 module to form the SPD-Conv module. (2). We add the SE attention module [15] in the Backbone part of the YOLOv5 model.

Finally, experimental results demonstrate that our proposed model outperforms other target detection models in terms of metrics and effectively detects small and edge targets in low-resolution images.

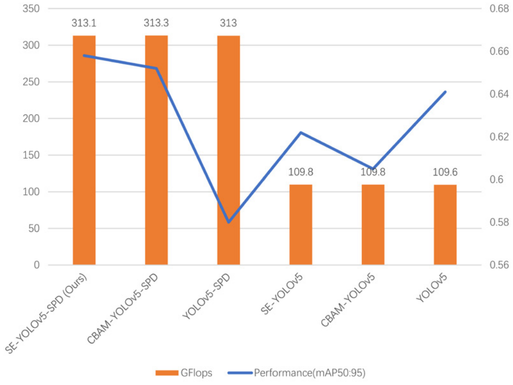

Figure 2. Improvement of our proposed approach over other YOLOv5-based models

The main contributions of this paper are as follows:

(1) We build a dataset by collecting log images from natural scenes, train the models, and compare them to the baseline models to validate their effectiveness.

(2) The effectiveness of the SE mechanism module and the SPD-Conv module is evaluated through comparative experiments.

(3) A new SE-YOLOv5-SPD logs detection model is proposed and its performance is evaluated. Figure 2 shows the performance and complexity compared with other YOLOv5-based models.

2 Related Works

2.1 Object Detection for Logs Detection

With the global objectives of reaching a carbon peak and achieving carbon neutrality, the intelligent use of forest resources plays a crucial role in reducing energy consumption and lowering greenhouse gas emissions. Forests hold immense value, which makes it all the more important to ensure that the utilization of forestry resources such as logs is done sustainably and efficiently. In daily wood processing operations, efficient management of incoming and outgoing storage is a crucial task prior to processing. However, traditional manual inspection methods can be expensive, inefficient, and prone to errors. An effective solution to overcome these challenges is to employ computer vision techniques for detecting logs.

As computer vision technology has advanced, the field of log detection has become increasingly diverse, with two main categories emerging: machine learning-based and deep learning-based methods.

Machine learning-based methods utilize statistical models to detect objects within an image, such as random forest model and PWL-Haarcascades model [1-2]. On the other hand, deep learning-based methods, use neural networks to learn complex features of objects, enabling them to accurately detect objects in images with varying levels of complexity. Examples of deep learning-based models include YOLOv4 and SSD model [3-5].

The field of deep learning has seen significant advancements in recent years, leading to the development of many detection models, such as YOLO family [6-9]. In this paper, we focus on improving the YOLOv5 model to address two specific challenges in logs detection: detecting logs in low-resolution images and detecting small objects at the image edge.

2.2 Improved YOLOv5 Models with Different Tricks

The YOLOv5 model is a popular object detection model known for its excellent detection capabilities, modularity, and scalability.

Several improvements have been made to the model by incorporating attention mechanism modules. B. Yan et al. added three SENet attention modules to YOLOv5 layers to improve the real-time detection of apples [10]. J. Qi et al. added a SE-Res attention mechanism module to detect tomato leaves virus [12]. S. Li used the ECA module to improve the detection of plant seeds [11]. L. Yang et al. and R. Li et al. used the CBAM module to detect the field flat jujube and wheat ears [13-14]. These improvements have shown significant advancements in object detection.

The YOLOv5 model and its improved versions may face limitations in detecting small and complex objects, especially those located at the edges of images or in low-resolution settings. To address this challenge, R. Sunkara et al. [17] have proposed the SPD-Conv module, which uses the SPD module to downsample feature maps and a non-stride convolution module to retain all discriminative feature information. This approach can improve detection for low-pixel images and small targets. However, using the SPD-Conv module alone may not yield optimal results, as shown in Figure 2 and Table 3. To overcome this limitation, we have combined the SPD-Conv module with the SE module and integrated them into the YOLOv5 model. This solution addresses the challenge of detecting small objects in low-pixel images and enhances the global detection performance of the YOLOv5 model.

Overall, deep learning is a promising approach to the problem of log detection. In particular, single-stage models like YOLOv5 have shown good detection performance and scalability in engineering. Moreover, improved models based on YOLOv5 have achieved impressive results in many scenarios. However, current object detection models do not have a good solution for some issues, such as low-pixel input images and small targets at the edges of the image. Therefore, we propose the SE-YOLOv5-SPD model to address these problems.

3 Data Acquisition and Dataset Construction

This section mainly introduces the dataset used in the experiment, as well as data collection and selection, pre-processing, and data labeling.

3.1 Criteria and Methods for Dataset Selection

To verify the robustness of the model under different environmental disturbances, we capture images of stacked logs and other complex situations such as tree branch shading, cluttered stacked logs, and high light or low light conditions. Moreover, we adjust the distance and angle to make our dataset have more different image edges, which can improve our training models to detect the small-size objects in image edges.

3.2 Data Acquisition Process and Final Selected Dataset

The dataset in this paper was collected on July 27th, 2022 and January 19th, 2022 from the timber market in Xixia District, Yinchuan, Ningxia, China and the forest farm near Shishou No. 1 Middle School, Hubei, China. Finally, 363 images were collected, including 304 images with 4896×3672 pixels and 59 images with 3648×1680 pixels.

We deleted images that had similar angles, poor shooting effects (such as overexposure), and overly tilted angles. In the end, 145 images are finally retained for training. Among them, 122 were used as training data, and 23 are used for validation.

3.3 Labeling of Images

The datasets used in this paper are all annotated using Labelme annotation software to generate JSON files in labelme format. We convert the annotation information into VOC format required for Faster R-CNN model [18], and SSD model [4], and YOLO format [9] required for YOLOv5 model.

3.4 Enhancement of Data

In this work, we faced challenges such as uneven lighting, occlusion from tree branches and leaves, and variations in input image resolution. To address these issues, we standardized input image resolution to 640×640 in our experiments. To improve the model's training performance, we applied data enhancement techniques using the Albumentations and Mosaic packages. The Mosaic method enhances data by combining one central image with three random images, randomly placed within the main image, and cropping them into one image. The Albumentations package enhances data by applying filters and grayscale processing.

4 Methodology

This section will introduce the structure of SE-YOLOv5-SPD model involved in this paper.

4.1 YOLOv5 Model

YOLOv5 model is a popular and accurate one-stage target detection model. It consists of three parts (Backbone, Neck, and YOLO Head), and some preprocessing (e.g., mosaic data enhancement, automatic anchoring, and adaptive image scaling) and normalization operations are added at the beginning of the whole network.

Before the images are input to the YOLOv5 model, some pre-processing is needed, such as data enhancement, image cropping and stitching, and pixel transformation to the appropriate size. Then the model needs to calculate the most appropriate anchors for the dataset. Firstly, the image data is randomly scaled, cropped, and aligned by Mosaic data augmentation, which not only expands the data but also allows the model to train multiple image data at one time. Secondly, this paper also uses the image enhancement function of Albumentations package along with Mosaic data enhancement, which can add some filters to the images and change the image quality so that the training data set can be added to the original data set. Then, for the resolution of the input image of this model, the input image pixel is set to 640×640 pixels because of the problem of non-uniformity of input image pixels in the future and also to solve the target detection problem well with low pixel images. The autoanchor algorithm in the model is used to automatically calculate the suitable anchors of the dataset to minimize the subsequent detection error.

As previously mentioned, the YOLOv5 model is divided into three parts, Backbone, Neck, and YOLO Head.

The Backbone part uses a combination of multiple Conv modules and C3 modules, and finally through the structure of the SPPF module. Among them, the Conv modules are all convolution modules with a kernel of 3, stride of 2, and no pooling operation. The C3 module is based on the design idea of CSPNet, using two ResNet modules on three convolutional layers to perform residual connections. Its function is to extract information during the convolution operation while avoiding gradient information duplication in the network learning process, so that the accuracy will not decrease while the network is lightweight. The SPPF module is to first perform a convolution operation with a kernel of 1, the stride of 1 on the input data to reduce the number of input data channels by half, and then through three maxpooling and concatenating the results of these three maxpooling.

The Neck section of the network utilizes a FPN+PAN structure to simultaneously upsample and downsample image information [19-20], allowing for the extraction of both high-dimensional and low-dimensional features.

The YOLO Head part is made up of three Head structures with a 1x1 convolutional structure. It is used to link the outputs of the last three C3 modules of the neck and to produce these high-resolution frames with varying sampling intervals.

The YOLOv5s model of the YOLOv5 series is selected in this paper, which has a simple model architecture and excellent detection effect and is very suitable for the problem scenarios. Figure 3 illustrates the network structure of the YOLOv5 model.

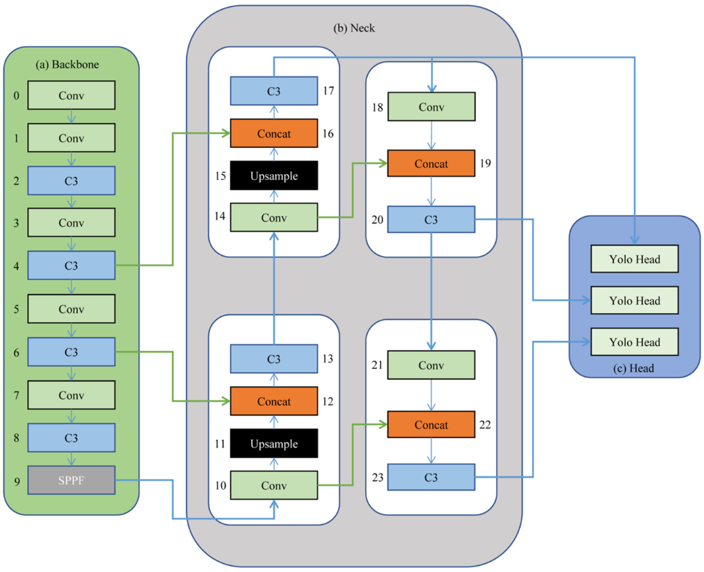

Figure 3. YOLOv5 model

4.2 SPD-Conv Module and Focus Module

The convolution module in CNN extracts the features of image data by using convolution and max-pooling, and then the CNN filter will generates the feature maps. They obtain the high-dimensional information of the image. However, their performance will decrease when the resolution is low, or the task target is small because some information is inevitably lost in this series of operations.

In this paper, we use a new convolution method called SPD-Convolution, which extracts the features of the image in two steps, Space-to-depth (SPD) module and Non-strided Convolution, to reduce the loss of information and finally optimize the problem of low resolution and small task target.

4.2.1 Space-to-Depth (SPD) Module

The SPD Module plays a crucial role in downsampling the feature maps within and across a CNN [17]. The module operates by slicing and concatenating the feature maps.

Given a feature map $X_{i,j}$ of size $\boldsymbol{S} \times \boldsymbol{S} \times \boldsymbol{C}_1$ from the previous layer, a sequence of sub-feature maps $f_{x,y}$ is extracted. These sub-feature maps $f_{x,y}$ are slicing by feature maps $X_{i,j}$ such that the $i + x$ and $j + y$ are divisible by scale, and $x + y \ = scale^2$ (The scale is a hyperparameter assigned prior to training, determines the range of the feature map slices). Each sub-feature map $f_{n,m}$ which is shape of ($\frac{S}{scale}$, $\frac{S}{scale}$, $scale^2C_1$). The calculation process of sub-feature maps $f_{x,y}$ is described as follows:

$$\begin{aligned} &f_{0,0} = X\,[0\colon S\colon scale,\ 0\colon S\colon scale] \\ &f_{1,0} = X\,[1\colon S\colon scale,\ 0\colon S\colon scale] \\ &\dots \\ &f_{scale-1,0} = X\,[scale-1\colon S\colon scale,\ 0\colon S\colon scale] \end{aligned} \quad \textbf{(1)}$$

$$\begin{aligned} &f_{0,1} = X\,[0\colon S\colon scale,\ 1\colon S\colon scale] \\ &f_{1,1} = X\,[1\colon S\colon scale,\ 1\colon S\colon scale] \\ &\dots \\ &f_{scale-1,1} = X\,[scale-1\colon S\colon scale,\ 1\colon S\colon scale] \end{aligned} \quad \textbf{(2)}$$

$$\dots\dots$$

$$\begin{aligned} &f_{0,scale-1} = X\,[0\colon S\colon scale,\ scale-1\colon S\colon scale] \\ &f_{1,scale-1} = X\,[1\colon S\colon scale,\ scale-1\colon S\colon scale] \\ &\dots \\ &f_{scale-1,\,scale-1} = X\,[scale-1\colon S\colon scale,\ scale-1\colon S\colon scale] \end{aligned} \quad \textbf{(3)}$$

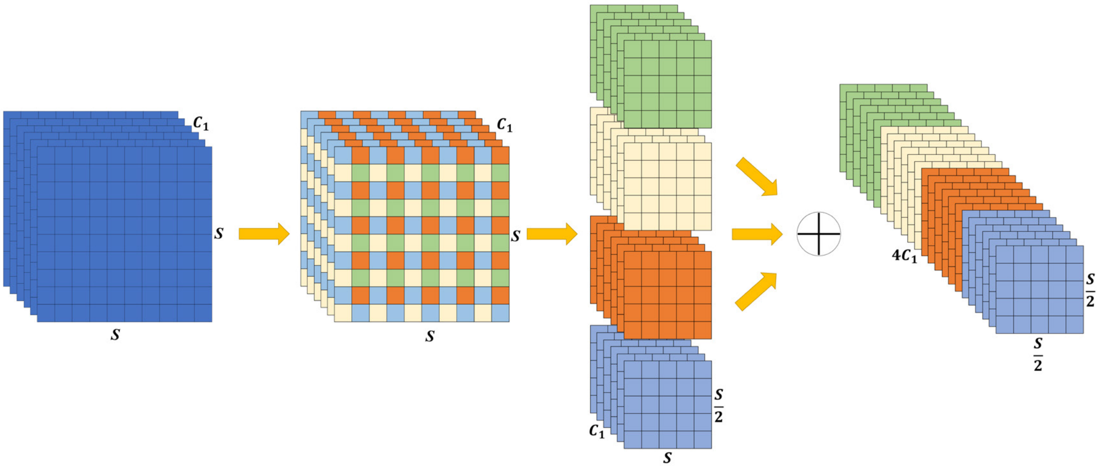

Figure 4. Space-to-Depth (SPD) module

Note. The orange arrows represent the transformation process of the data; S, S, C_1 represent the tensor shape (height, width, depth) of feature maps; The cross icon operation is concatenated.

The first three steps of Figure 4 show the feature map slices (scale=2), where the feature maps X is sliced to four sub-feature maps $f_{0,0}, f_{0,1}, f_{1,0}, f_{1,1}$. Each sub-feature maps have the shape of $(\frac{S}{2}, \frac{S}{2}, C_1)$.

In the next step, the sub-feature maps $f_{x,y}$ will concatenate to a new sub-feature maps $X'(\frac{S}{scale}, \frac{S}{scale}, scale^2C_1)$ by patching the each sub-feature map. The end step of Figure 4 shows the four sub-feature maps $f_{0,0}, f_{0,1}, f_{1,0}, f_{1,1}$ concatenate to the sub sub-feature maps $X'(\frac{S}{2}, \frac{S}{2}, 2^2C_1)$.

The SPD operation on feature map X allows the model to maximize the sampling of the information in X. The segmentation operation in the module allows the model to focus more on the small targets in the image.

4.2.2 Non-strided Convolution Module

After applying the SPD feature upsampling module, we utilize the non-strided convolution module to preserve as much discriminative feature information as possible.

For any X' is convolved by a Non-strided Convolution layer with C_2 filters where $\boldsymbol{C_2 < scale^2C_1}$, then let sub-features map $X'(\frac{S}{scale}, \frac{S}{scale}, scale^2C_1)$ transforms to $X''(\frac{S}{scale}, \frac{S}{scale}, C_2)$. Figure 5 shows the Nonstrided Convolution Module.

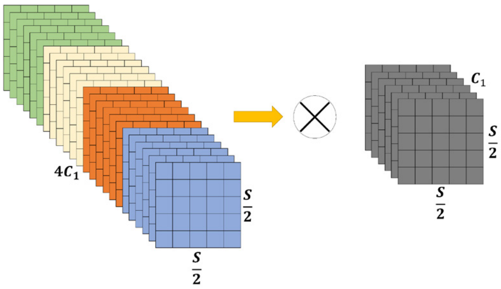

Figure 5. Non-Strided convolution module

Note. The S or C_1 next to the feature maps represents the tensor shape (height, width, depth). The X icon represents multiply and Convolution.

4.2.3 SPD-Conv Module

This paper follows the Non-strided Convolution, a convolution module with kernel=1, stride=1, and without Pooling operation.

In the experiments [18], the SPD-Conv module uses the SPD module to collect the Feature maps and then passes through the C3 module in YOLOv5. Lastly, through the nonstrided convolution to complete the SPD-Convolution operation.

We are also using this architecture in the SE-YOLOv5-SPD model, so the purpose is to maximize the performance of the SPD module after the C3 module and non-stride convolution module, compared with directly connecting the SPD module to the nonstrided convolution module.

4.2.4 Focus Module

The official document of YOLOv5 uses a focusing module composed of an SPD module directly connected to the non-staggered convolution. Because SPD and non-staggered convolution modules can effectively use CPU or GPU computing resources to achieve efficient computation, this module accelerates the model's training.

In this paper, the Focus module is applied to the YOLOv5-Focus and SE-YOLOv5-Focus models, replacing all the SPD-Conv modules of YOLOv5-SPD and SE-YOLOv5-SPD to validate the effect of the SPD-Conv module.

4.3 Squeeze-and-Excitation Networks (SENet) Block

Due to the complexity of the environment in this dataset and the limited number of training data, as well as the low resolution of the images used in the article and the small size of the objects to be detected among many trees. In addition, the dense sampling of SPD-Conv on the image makes it difficult to obtain good training results with the original network architecture.

Thus, this paper also introduces the attention mechanism of the SE channel to improve the YOLOv5 model. Adding the SE attention mechanism module between the Backbone and Neck can make the network pay more attention to the target to be detected before upsampling and after the intensive sampling of the SPD-Conv module and improve the detection effect. Figure 6 shows the SE Block.

4.4 SE-YOLOv5-SPD Model

By adding SENet block and SPD-Conv module, we proposed the SE-YOLOv-SPD model. We are using this model to solve the problem of a complex environment and small detection objects.

This paper added the SPD module before each C3 module in the YOLOv5 model to solve the impact of low-resolution images on the detection results. It turns the subsequent Conv modules into stride as one to form the SPD-Conv module, but the intensive sampling of SPD-Conv module may make it difficult for the YOLOv5 model to locate the target area. Therefore, we add the SE attention mechanism module to the back of the Backbone module in the YOLOv5 model to solve this problem. Finally, we obtain a SE-YOLOv5-SPD model that works well in the face of the issues above (Detection of log ends in environments with low image pixels and complex presence). Figure 7 illustrates the network structure of the SE-YOLOv5-SPD model.

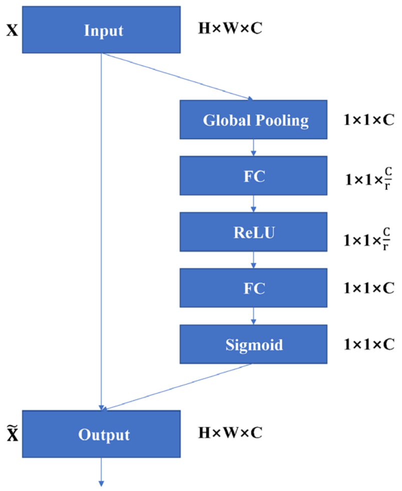

Figure 6. SENet block

Note. H, W, C represents the tensor shape (height, width, depth) of data; FC represents fully connected neural network.

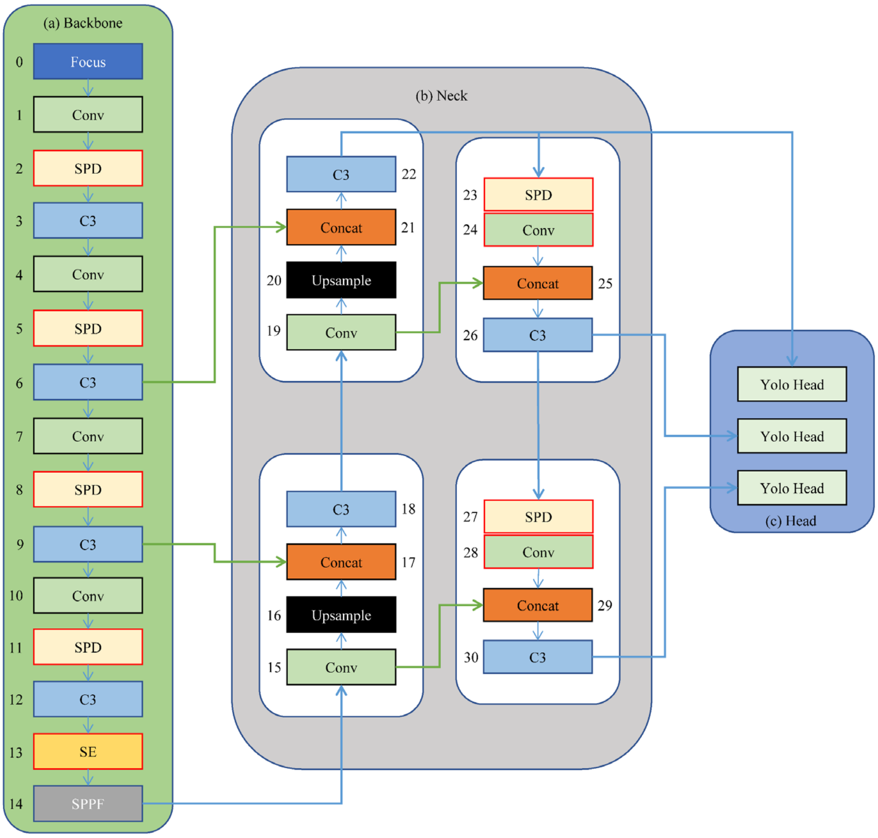

Figure 7. SE-YOLOv5-SPD model

4.5 Pre-Trained Models

The pre-trained model is a set of network weights that some researchers share for others to use after the model has been trained with better training results. This also allows the training to continue in a model with good network weights, which helps the training of a model that is more suitable for the task under a model with better results.

The pre-trained models utilized in this study were constructed using the PyTorch framework, with all network weights sourced from either PyTorch or YOLOv5 official. Nonetheless, certain models, such as CBAM-YOLOv5 and YOLOv5-Focus, lacked appropriate pre-trained models. Therefore, we adopted the same pre-trained model training strategy outlined in the YOLOv5 documentation, which involves training on the COCO128 dataset comprising the first 128 images of the COCO2017 dataset for 300 epochs. Following training, the pre-trained model exhibiting the most promising outcomes was selected for this experiment.

5 Result and Discussion

5.1 Evaluation Metrics

In this paper, four evaluation metrics of detection results are selected: Precision, Recall, mAP_{50} and $mAP_{50:95}$, where Precision is the ratio of the number of correctly predicted images to the total number of positive class predictions. Recall is the number of positive class predicted images to all labeled images. Among them, mAP_{50} and $mAP_{50:95}$ are two kinds of mAP (Mean Average Precision), mAP_{50} is the average value of Precision with $IOU \geq 0.50$. $mAP_{50:95}$ is calculated by averaging the precision values for all IOU thresholds ranging from 0.5 to 0.95 with a step of 0.05. IOU (Intersection over Union) is a metric that is used to measure the overlap between the predicted result and the ground truth. It calculates the ratio of the area of overlap between the two to the total area of union. The four evaluation metrics mentioned above are calculated as follows.

$$Precision = \frac{TP}{TP + FP}. \tag{4}$$

$$Recall = \frac{TP}{TP + FN}. \tag{5}$$

$$mAP_{50} = \frac{\sum_i^K AP_i(IOU \geq 0.5)}{K}. \tag{6}$$

$$mAP_{50:95} = \frac{\sum_i^K AP_i(IOU \geq 0.5 \ to \ 0.95)}{K}. \tag{7}$$

In equation (4), TP is the true result of the predicted positive class, FP is the true result of the predicted negative class, and TP+FP is the result of all Positive images, i.e. the number of images in the predicted positive class.

In equation (5), TP is the number of predicted positive images that are also positive, FN is the number of predicted positive images that are negative, and TP+FN is the number of images that fully satisfy the image annotation.

In equation (6) and equation (7), K is the number of classes in the dataset, but since the task of this paper belongs to one-class object detection, K=1 in the two metrics of mAP_{50} and $mAP_{50:95}$. And the $\Sigma_i^K AP_i$ and IOU are calculated as follows:

$$IOU = \frac{Area \ of \ Overlap}{Area \ of \ Union}. \tag{8}$$

$$\sum_i^K AP_i = \int_0^1 Precision \ (Recall) \ dRecall. \tag{9}$$

We use the GFLOPs (Giga floating-point operations) metrics to measure each model’s complexity. FLOPs is a floating point arithmetic numbers. Can be used to measure the complexity of the algorithm/model. A GFLOPS (gigaFLOPS) is equal to one billion (10^9) floating-point operations.

5.2 Experimental Environment and Training Configuration

All models involved in this paper are trained and validated on Ubuntu 18.04.3 server with Intel(R) Xeon(R) Gold 6140 CPU @ 2.30 GHz and NVIDIA Tesla V100 SXM2 32 GB on the system is Ubuntu 18.04.3.

All the models involved in this paper were built in Pytorch, a deep learning framework based on Python.

The input image pixels of all the models involved in this paper are 640×640 pixels, the Batch size is 16; the Epoch of training is 300, the Learning rate of the pre-trained models is 0.01, and the Learning rate of all other models is 0.025. The learning rate of all models is decremented by 0.01 every 50 epochs to prevent the model from over-fitting, and the model optimizer is selected as Adam optimizer [21-23].

5.3 Compare the Log Detect Results of Different Models

In this section, SE-YOLOv5-SPD is compared with other models (YOLOv5, Faster R-CNN, and SSD) with excellent results on the same dataset, and the following results are finally obtained (the underlined model is the model proposed in this paper and has not been used in any articles).

It can be seen from Table 1 that SE-YOLOv5-SPD is the model with the best performance, where the results of $mAP_{50:95}$ is 0.652, mAP_{50} is 0.912, Precision is 0.968, and Recall is 0.864. The results show that SE-YOLOv5-SPD performs well, and SE-YOLOv5-SPD has good logarithmic detection ability.

Moreover, compared with the Faster R-CNN and SSD models, the two YOLOv5-based models have better detection results, demonstrating the power of the YOLOv5 model architecture for the target detection task. The detection results of the models in this paper are improved after adding the

SE Block and SPD-Conv modules, which also indicates that YOLOv5 has high scalability and can be further optimized and enhanced.

In Figure 8 to Figure 15, we show the visual comparison of the detection results of SE-YOLOv5-SPD with SE-YOLOv5, YOLOv5, and Faster R-CNN (Figure 8 and Figure 12 are SE-YOLOv5-SPD model detection result, Figure 9 and Figure 13 are SE-YOLOv5 model detection result, Figure 10 and Figure 14 are YOLOv5 model detection result, Figure 11 and Figure 15 are Faster R-CNN model detection result). We can see from Figure 7 to Figure 14 that SE-YOLOv5-SPD not only has excellent results in the overall recognition but also can have good detection of the smaller log ends at the edge of the image. Moreover, we found that the SE-YOLOv5 and YOLOv5 models have good general results in detecting log ends but can not achieve good recognition when facing small log ends. However, the Faster R-CNN model did not perform well for detection in this work.

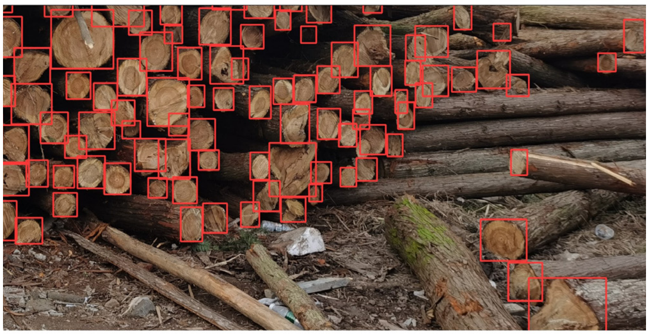

Figure 8. SE-YOLOv5-SPD model detection results 1

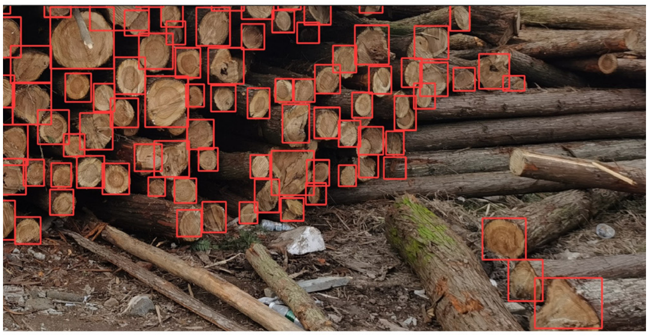

Figure 9. SE-YOLOv5 model detection results 1

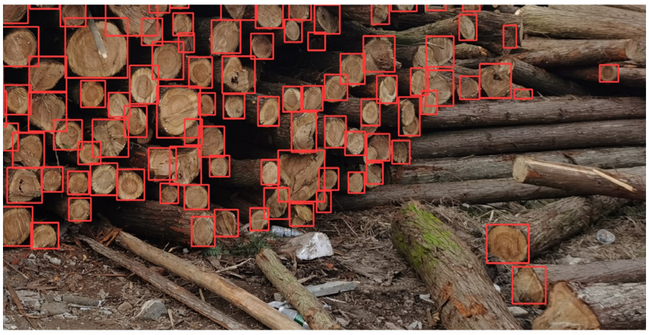

Figure 10. YOLOv5 model detection results 1

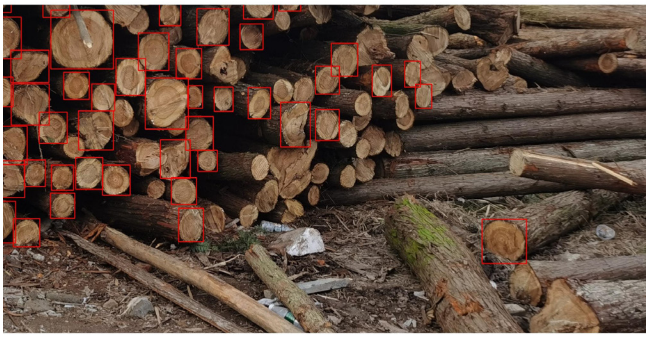

Figure 11. Faster R-CNN model detection results 1

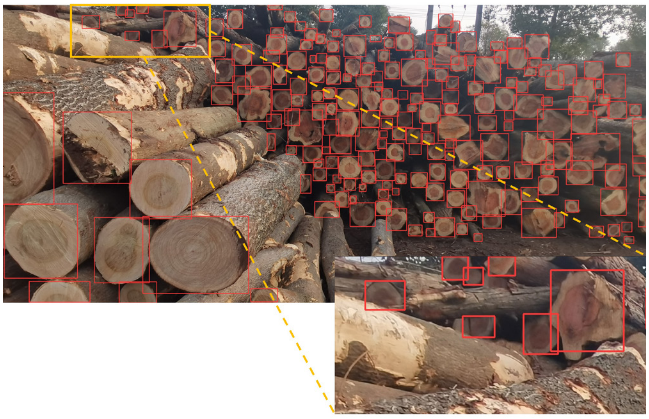

Figure 12. SE-YOLOv5-SPD model detection results 2

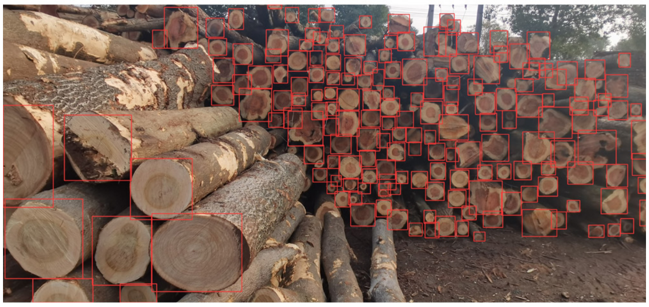

Figure 13. SE-YOLOv5 model detection results 2

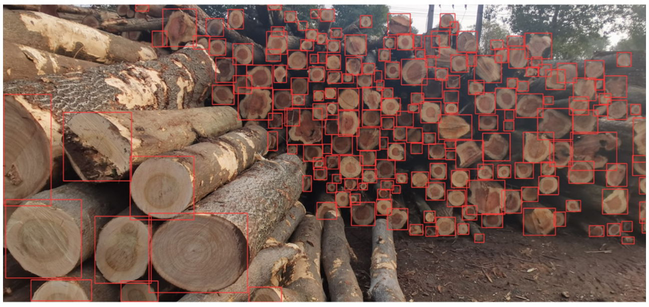

Figure 14. YOLOv5 model detection results 2

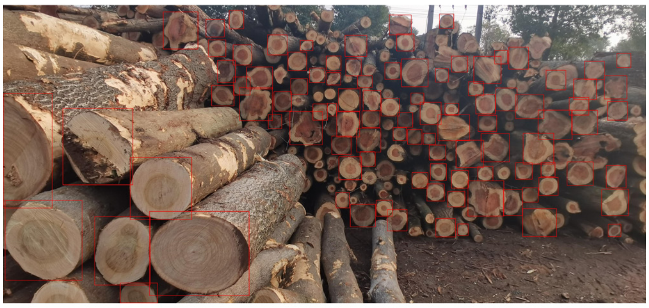

Figure 15. Faster R-CNN model detection results 1

5.4 Comparison of the Effects of Different Attention Mechanisms on the Detection Effect of Logs End

In this section, we experimentally compare the SE-YOLOv5-SPD with the CBAM attention mechanism, and we replace the SE attention module in the original model with the CBAM attention module. We also verify the different effects of SPD-Conv module with other attention modules and the effects of the attention mechanism model without the SPD-Conv module. The results are shown in Table 2 (the underlined model is the model proposed in this paper and has not been used in any papers).

The result shows that SE-YOLOv5-SPD still plays an excellent effect compared to no attention mechanism. It is worth noting that the detection of the model is not as good as the initial YOLOv5 model with the addition of the attention module alone. In addition, the detection of the model is significantly improved with the addition of the attention module.

5.5 Analyse the Function of the SPD-Conv Module

In this section, we experimented with the Focus module, which has a similar structure to the SPD-Conv module, and the effect of adding an SE attention mechanism is tested. The results are shown in Table 3 (the underlined model is the model proposed in this paper and has not been used in any papers).

From the experimental results, we know that the SE-YOLOv5-SPD model is still the best-performing model, and the model after the SPD-Conv module is more effective than the Focus module. Although the SE-YOLOv5-Focus model has some improvement after adding the SE Attention module, it still can not be compared with the YOLOv5 model and the model with the Attention mechanism and SPD-Conv module. This experiment also demonstrates that the SPD-Conv module plays a vital role in solving this task.

It is also noteworthy that, similar to the detection results when only adding the attention mechanism to the model. If only add the SPD-Conv module to the model, the detection results do not get better but decrease. However, after adding both the attention mechanism and the SPDConv module, the detection effectiveness of the model is greatly improved.

The results in Table 1 show that the SE-YOLOv5-SPD model has a significant advantage compared with different classical models, which proves the excellent effectiveness of the SE-YOLOv5-SPD model in solving the tasks in this paper. Figure 7 and Figure 8 show that it is challenging to play a significant role when only the attention module or SPD-Conv module is added. However, by complementing the attention module and SPD-Conv module, the network structure can perform the model well and significantly improve the model's detection.

Table 1. Different models' evaluation results

Models	GFLOPs	$mAP_{50:95}$	mAP_{50}	Precision	Recall
SE-YOLOv5-SPD	313.1	**0.658**	**0.915**	**0.980**	**0.867**
SE-YOLOv5	109.8	0.622	0.894	0.970	0.840
YOLOv5	109.6	0.641	0.904	0.966	0.849
Faster R-CNN	-	0.521	0.701	-	-
SSD	-	0.538	0.795	-	-

Table 2. Evaluation results of different attention mechanism models

Models	GFLOPs	$mAP_{50:95}$	mAP_{50}	Precision	Recall
SE-YOLOv5-SPD	313.1	**0.658**	**0.915**	**0.980**	**0.867**
CBAM-YOLOv5-SPD	313.3	0.652	0.913	0.947	0.866
SE-YOLOv5	109.8	0.622	0.894	0.970	0.840
CBAM-YOLOv5	109.8	0.605	0.890	0.967	0.834
YOLOv5	109.6	0.641	0.904	0.966	0.849

Note. The model containing the CBAM module is produced by replacing the SE module.

Table 3. Evaluation results of SPD-Conv module and Focus module

Models	GFLOPs	$mAP_{50:95}$	mAP_{50}	Precision	Recall
SE-YOLOv5-SPD	313.1	**0.658**	**0.915**	**0.980**	**0.867**
CBAM-YOLOv5-SPD	313.3	0.652	0.913	0.947	0.866
SE-YOLOv5-Focus	177.7	0.584	0.874	0.950	0.815
YOLOv5-SPD	313	0.580	0.879	0.963	0.812
YOLOv5-Focus	177.6	0.497	0.826	0.941	0.744
YOLOv5	109.6	0.641	0.904	0.966	0.849

Note. The model containing the Focus module is produced by changing the position of the Non-stride Convolution module in the SPD-Conv module.

6 Conclusion

In this paper, we propose a log detection method using SE-YOLOv5-SPD to address the challenges of detecting logs in farm scenes, where the arrangement is untidy and object sizes are inconsistent. The performance of the SE attention module and SPD-Conv module are also compared to the CBAM attention module and Focus module within the SE-YOLOv5-SPD model.

The experimental evaluation results show that the performance of SE-YOLOv5-SPD model is better than the other baseline models, the metrics of $mAP_{50:95}$ results in 0.652, mAP_{50} in 0.912, Precision in 0.968, and Recall in 0.864. The detection results of other models also show that the SE-YOLOv5-SPD model has the best performance. This is primarily due to the implementation of the SPD-Conv module, which enhances the model's downsampling information, preserves discriminative feature information through non-strided convolution, and simplifies network training with the SENet attention mechanism.

This research presents a solution for detecting stacked logs in forestry fields and highlights the need for a relevant dataset to improve the accuracy of the detection algorithm.

In future work, there are several directions to improve and explore for better efficiency and accuracy in detecting logs in wood processing plants using the SE-YOLOv5-SPD model. Firstly, the model's adaptability to different environments such as forests and logging sites can be expanded by training it with data from these new settings. Secondly, increasing the amount and diversity of training data can improve the model's accuracy and generalization ability. Thirdly, compressing the model's size can allow it to be deployed on mobile devices or microcontrollers for more efficient detection. Finally, enhancing the model's generalization ability through architecture optimization, loss function refinement, and regularization can improve its ability to adapt to new scenarios and improve performance overall.

Acknowledgment

This work was supported by grants from the Natural Science Foundation Project of Ningxia Province, China titled "User-oriented Multi-criteria Relevance Ranking Algorithm and Its Application (2021AAC03205)", Key R&D Program for Talent Introduction of Ningxia Province China titled "Research on Key Technologies of Scientific Data Retrieval in the Context of Open Science (2022BSB03044)", Starting Project of Scientific Research in the North Minzu University titled "Research of Information Retrieval Model Based on the Decision Process (2020 KYQD37)" and North Minzu University Postgraduate Innovation Project (YCX22178, YCX 22193).

References

[1] Y. V. Chiryshev, A. S. Atamanova, *Automatic wood log detection based on random decision forests learning algorithm and histogram of oriented gradients*, August, 2017, https://ceur-ws.org/Vol-1909/paper2.pdf.

[2] E. Gutzeit, J. Voskamp, Automatic segmentation of wood logs by combining detection and segmentation, *8th International Symposium on Visual Computing*, Crete, Greece, 2012, pp. 252-261.

[3] H. Tang, K. Wang, J. Gu, X. Li, W. Jian, Application of ssd framework model in detection of logs end, *Journal of Physics: Conference Series*, Vol. 1486, Article No. 072051, 2020.

[4] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, A. C. Berg, Ssd: Single shot multibox detector, *14th European Conference on Computer Vision*, Amsterdam, The Netherlands, 2016, pp. 21-37.

[5] Y. Lin, R. Cai, P. Lin, S. Cheng, A detection approach for bundled log ends using k-median clustering and improved yolov4-tiny network, *Computers and Electronics in Agriculture*, Vol. 194, Article No. 106700, March, 2022.

[6] J. Redmon, S. K. Divvala, R. B. Girshick, A. Farhadi, You only look once: Unified, real-time object detection, *29th IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016, pp. 779-788.

[7] J. Redmon, A. Farhadi, *Yolov3: An incremental improvement*, April, 2018, https://arxiv.org/pdf/1804.02767.pdf.

[8] A. Bochkovskiy, C. Wang, H. M. Liao, *Yolov4: Optimal speed and accuracy of object detection*, April, 2020, https://arxiv.org/pdf/2004.10934v1.pdf.

[9] Github, *YOLOv5 Classification Models*, https://github.com/ultralytics/yolov5/, August, 2022.

[10] B. Yan, P. Fan, X. Lei, Z. Liu, F. Yang, A real-time apple targets detection method for picking robot based on improved yolov5, *Remote Sensing*, Vol. 13, No. 9, Article No. 1619, May, 2021.

[11] S. Li, S. J. Zhang, J. Xue, H. X. Sun, R. Ren, A fast neural network based on attention mechanisms for detecting field flat jujube, *Agriculture*, Vol. 12, No. 5, Article No. 717, May, 2022.

[12] J. Qi, X. Liu, K. Liu, F. Xu, H. Guo, X. Tian, M. Li, Z. Y. Bao, Y. Li, An improved yolov5 model based on visual attention mechanism: Application to recognition of tomato virus disease, *Computers and Electronics in Agriculture*, Vol. 194, Article No. 106780, March, 2022.

[13] L. Yang, J. Yan, H. Li, X. Cao, B. Ge, Z. Qi, X. Yan, Real-time classification of invasive plant seeds based on improved yolov5 with attention mechanism, *Diversity*, Vol. 14, No. 4, Article No. 254, April, 2022.

[14] R. Li, Y. Wu, Improved yolo v5 wheat ear detection algorithm based on attention mechanism, *Electronics*, Vol. 11, No. 11, Article No. 1673, June, 2022.

[15] J. Hu, L. Shen, G. Sun, E. Wu, Squeeze-and-excitation networks, *31st IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 7132-7141.

[16] S. Woo, J. Park, J. Lee, I. Kweon, Cbam: Convolutional block attention module, *15th European Conference on Computer Vision*, Munich, Germany, 2018, pp. 3-19.

[17] R. Sunkara, T. Luo, *No more strided convolutions or pooling: A new cnn building block for low-resolution*

images and small objects, August, 2022, https://arxiv.org/pdf/2208.03641.pdf.

[18] S. Ren, K. He, R. Girshick, J. Sun, *Faster r-cnn: Towards real-time object detection with region proposal networks*, June, 2015, https://arxiv.org/pdf/1506.01497v1.pdf.

[19] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature Pyramid Networks for Object Detection, *30th IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017, pp. 936-944.

[20] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path Aggregation Network for Instance Segmentation, *31st IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 8759-8768.

[21] D. P. Kingma, J. Ba, *Adam: A method for stochastic optimization*, July, 2015, https://arxiv.org/abs/1412.6980v8.

[22] S. Wen, S. Xiao, Y. Yang, Z. Yan, Z. Zeng, T. Huang, Adjusting learning rate of memristor-based multilayer neural networks via fuzzy method, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 38, No. 6, pp. 1084-1094, June, 2019.

[23] S. Wen, H. Wei, Z. Yan, Z. Guo, Y. Yang, T. Huang, Y. Chen, Memristor-based design of sparse compact convolutional neural network, *IEEE Transactions on Network Science and Engineering*, Vol. 7, No. 3, pp. 1431-1440, July-September, 2020.

Biographies

Xianqi Deng is currently pursuing the B.E. degree in Computer Science and Technology with the College of Computer Science and Engineering, North Minzu University, China. His research interests include machine learning, deep learning, and computer vision.

Jianping Liu received the Ph.D. degree in information technology and digital agriculture from the Chinese Academy of Agricultural Sciences. He is currently a Lecturer in information sciences with the College of Computer Science and Engineering, North Minzu University, China. His research has been published in Library & Information Science Research, in 2019, Data Science Journal, in 2020, and IEEE Access, in 2023.

Cheng Peng received the B.E. degree in intelligent science and technology and Ph.D. degree in pattern recognition and intelligent system from Xidian University, Xi'an, China, in 2016 and 2022, respectively. He is a Postdoctoral Researcher with University of Florida. His research interests include remote sensing image processing, and EHRs information extraction.

Yingfei Wang is currently pursuing the master's degree with the College of Computer Science and Engineering, North Minzu University, China. Her research interests include interactive information retrieval and click model. Her research has been published in the International Conference on Cloud Computing and Intelligent Systems (CCIS 2022) and IEEE Access, in 2023.